Réparations de Bases de Données et Réponses Cohérentes aux Requêtes

Origines et Développements Ultérieurs

Hao ZHANG

Introduction

Origine des réparations de bases de données et de la réponse cohérente aux questions (CQA en anglais)

- Les réparations de bases de données et la réponse cohérente aux questions (CQA) sont nées d'un article de PODS (Principles of Database Systems) de 1999 traitant des bases de données incompatibles avec les contraintes d'intégrité (ICs en anglais).

Motivations pour les réparations de bases de données et CQA

- Motivé par la nécessité d'identifier et d'extraire des données cohérentes de bases de données violant des contraintes d'intégrité.

- L'objectif est de fournir une méthode permettant de répondre aux requêtes de manière cohérente pour toutes les données réparées possibles.

Définition des Réparations de Bases de Données

- Les réparations de bases de données impliquent des modifications d'une instance de base de données afin de garantir le respect d'un ensemble donné de contraintes d'intégrité.

- Les réparations sont généralement construites en insérant ou en supprimant des tuples, ou en modifiant les valeurs des attributs, afin de minimiser la différence par rapport à la base de données d'origine.

- L'objectif est de parvenir à un état cohérent dans lequel les contraintes d'intégrité sont satisfaites.

- Ces modifications doivent être minimales dans le cadre de l'inclusion d'un ensemble afin d'éviter les changements inutiles ; elles sont également connues sous le nom de réparations S (S-repairs en anglais).

Travaux Préliminaires

- L'accent a d'abord été mis sur des actions de mise à jour minimales pour restaurer la cohérence de la base de données tout en évitant des calculs de réparation exhaustifs non pratiques.

- Les premières recherches se sont étendues aux réparations de cardinalité, également appelées réparations C (C-repairs en anglais), et aux réparations basées sur la valeur nulle.

Comprendre les Réparations à l'Aide d'Exemples

Example 2.1. Consider the database instance D below, and the inclusion dependency

$$ID\colon\ \forall x\forall y\forall z(Supply(x,y,z)\ \rightarrow\ Articles(z)), \qquad (1)$$

requiring that the items shipped according to table *Supply* are all among the official list of items displayed in table *Articles*.

Supply	Company	Receiver	Item
	C_1	R_1	I_1
	C_2	R_2	I_2
	C_2	R_1	I_3

Articles	Item
	I_1
	I_2

This instance is inconsistent with respect to (wrt.) *ID*, i.e. it does not satisfy *ID*, usually denoted with $D \not\models ID$. Clearly there is a problem with the last tuple of table *Supply*, but the information in the other tuples of the database seems to be fine. Now, if we pose the query about the items that are supplied, i.e. the conjunctive query

$$Q(z)\colon\ \exists x\exists u Supply(x,u,z), \qquad (2)$$

Example 3.5. The following is a *denial constraint*, i.e. it prohibits ombinations or joins of database atoms:

$$\kappa\colon\ \neg\exists x\exists y(S(x)\wedge R(x,y)\wedge S(y)).$$

'he following database instance D violates κ.

R	A	B
ι_1	a_4	a_3
ι_2	a_2	a_1
ι_3	a_3	a_3

S	A
ι_4	a_4
ι_5	a_2
ι_6	a_3

Example 3.3. The following instance D violates the key constraint $KC\colon\ Name \rightarrow Salary$ that requires the employee salary to functionally depend upon the employee name, i.e. every employee should have at most one salary.

Employee	*Name*	*Salary*
	page	*5K*
	page	*8K*
	smith	*3K*
	stowe	*7K*

There are two repairs, D_1 and D_2 obtained by tuple deletions:

Employee	*Name*	*Salary*
	page	*5K*
	smith	*3K*
	stowe	*7K*

Employee	*Name*	*Salary*
	page	*8K*
	smith	*3K*
	stowe	*7K*

They are consistent with *KC* and minimally depart from D. Now, if we want the consistent answers to the query about all tuples in the database, i.e. to $Q_1(x,y)\colon\ Employee(x,y)?$, we obtain: $Cons(Q_1, D, \{KC\}) = \{\langle smith, 3K\rangle, \langle stowe, 7K\rangle\}$. Now, if the query is only about employee names, i.e. $Q_2(x)\colon\ \exists y Employee(x,y)?$, we obtain: $Cons(Q_2, D, \{KC\}) = \{\langle smith\rangle, \langle stowe\rangle, \langle page\rangle\}$ since we find *page* as employee in each repair. □

Sources: Carleton U. Ottawa U.

CUstds	Number	Name
	101	john
	102	mary

OUstds	Number	Name
	103	claire
	104	peter

SpecCU	Number	Field
	101	alg
	102	ai

SpecOU	Number	Field
	103	db

We are assuming the tables at the top have the student numbe as a key, and this constraint is satisfied, but this is not relevant fo the moment. The mediator has a single, global relation schema:

$$Stds(Number, Name, Univ, Field).$$

CC	AC	Phone	Name	Street	City	Zip
44	131	1234567	mike	mayfield	NYC	EH4 8LE
44	131	3456789	rick	crichton	NYC	EH4 8LE
01	908	3456789	joe	mtn ave	NYC	07974

The two following FDs

$$[CC, AC, Phone] \rightarrow [Street, City, Zip]$$
$$[CC, AC] \rightarrow [City]$$

are satisfied by D. They are "global" ICs that may not capture natural data quality requirements, e.g. as related to specific data values. Instead, the CFD

$$[CC = 44, Zip] \rightarrow [Street],$$

Autres Orientations de la Recherche sur les Réparations de Bases de Données

- Explorer les réparations basées sur les attributs, en particulier avec des valeurs numériques et des contraintes, pour relever de nouveaux défis.
- Étudier les réparations dans les bases de données spécialisées telles que les entrepôts de données, les bases de données spatiales et temporelles.

Lien entre la Causalité et les Réparations de Bases de Données

- La causalité explique pourquoi certains résultats de requête se produisent ou pourquoi des conditions sémantiques sont violées.
- Elle implique l'identification des tuples qui sont les causes contrefactuelles ou réelles d'un résultat de requête.
- Les causes et leurs responsabilités peuvent être dérivées des réparations de la base de données, ce qui permet de relier la causalité à la cohérence des données.

Conclusion

- Cet article retrace les origines et les motivations des réparations de bases de données et de la réponse cohérente aux requêtes (RCR), en mettant l'accent sur les premiers développements.
- Il examine l'évolution des concepts, y compris le document PODS de 1999 qui a jeté les bases de ce domaine.
- Les travaux soulignent l'importance d'une modification minimale des réparations et de la persistance de données cohérentes.
- Les orientations futures de la recherche sont également présentées, notamment l'exploration de la causalité dans les bases de données et l'application de la réparation dans différents domaines.